

Rapport oscilloscope

Jean Nanchen

December 20, 2020

HES-SO – 3èME ANNéE

Table des matières

[2. Introduction 2](#_Toc59397470)

[3. Création du projet 2](#_Toc59397471)

[4. Configuration du Timer 1 3](#_Toc59397472)

[5. Configuration de l’ADC 5](#_Toc59397473)

[6. Affichage 6](#_Toc59397474)

[7. Situation actuelle 6](#_Toc59397475)

[8. Questions 7](#_Toc59397476)

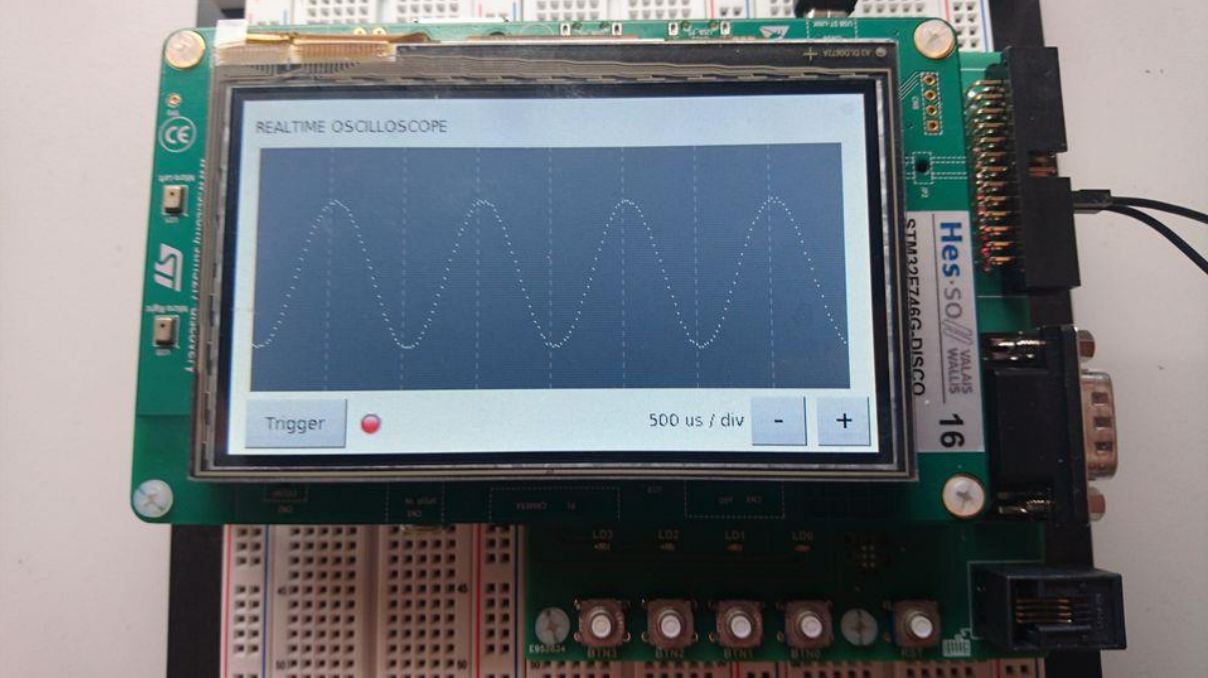
[Configuration du timing 7](#_Toc59397477)

[Configuration de l’ADC 8](#_Toc59397478)

[Sample-Rate Tuning 8](#_Toc59397479)

[9. Annexes 9](#_Toc59397480)

# Introduction

Le but de ce projet est d’échantillonner un signal à l’aide d’une entrée analogique et de l’afficher sur l’écran de notre discovery board.

# Création du projet

Le projet a d’abord été créé sur la base d’une configuration existante pour le discovery board.

De là, l’USB, le FatFS ainsi que l’Ethernet ont été désactiver.

Le RCC est configuré à 25MHz.

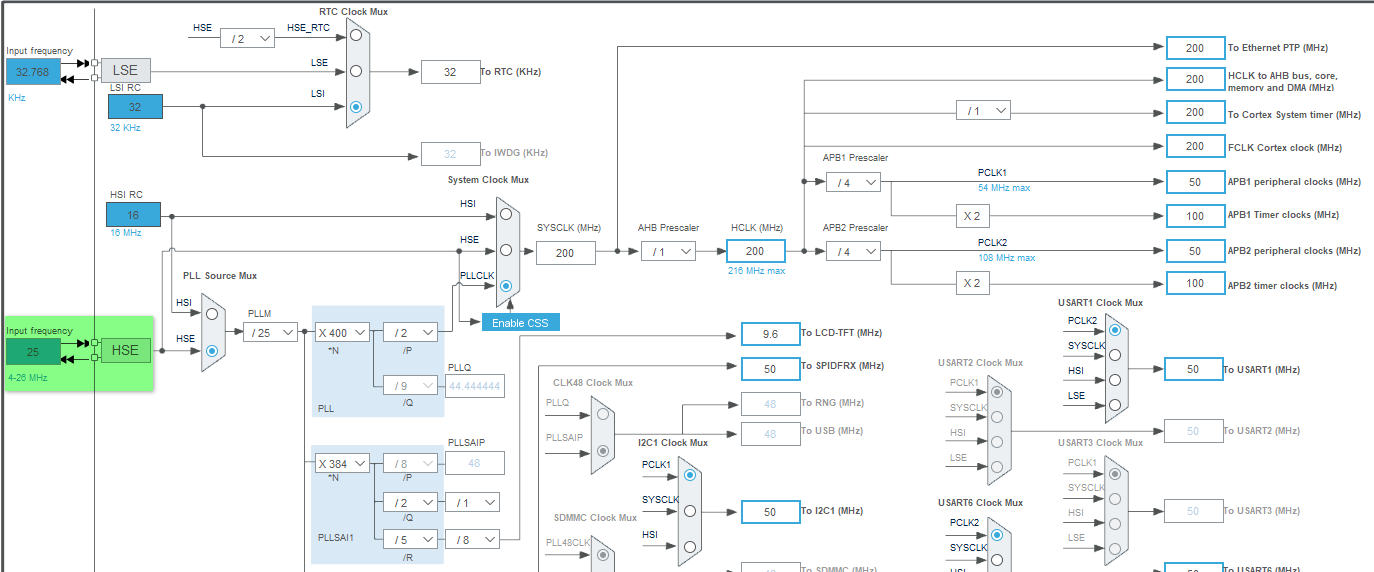
Les heap et le stack sont augmenté à 0x1000. L’IDE utilisé sera System Workbench.

Figure - Configuration du RCC

# Configuration du Timer 1

Le timer 1 est utilisé pour lancer les conversions AD.

Nous devons échantillonner des fréquences allant jusqu’à 1kHz. La fréquence d’échantillonnage (fs) doit être égale ou supérieur à :

Nous choisissons une fréquence d’échantillonnage de **10kHz** pour augmenter la précision.

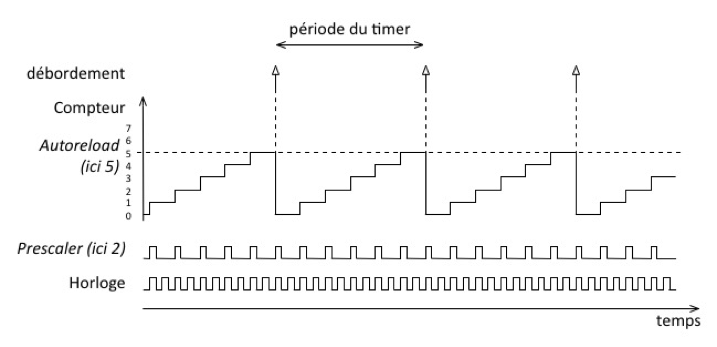
ABP2 est la clock qui incrémente le Timer 1. Il est cadencé à 100MHz (Figure 1). Pour générer un timeout chaque 10kHz, nous devons régler le préscaler à 100 et le counter period à 100.

Figure 2 - Préscaler & Counter Period (Autoreload)

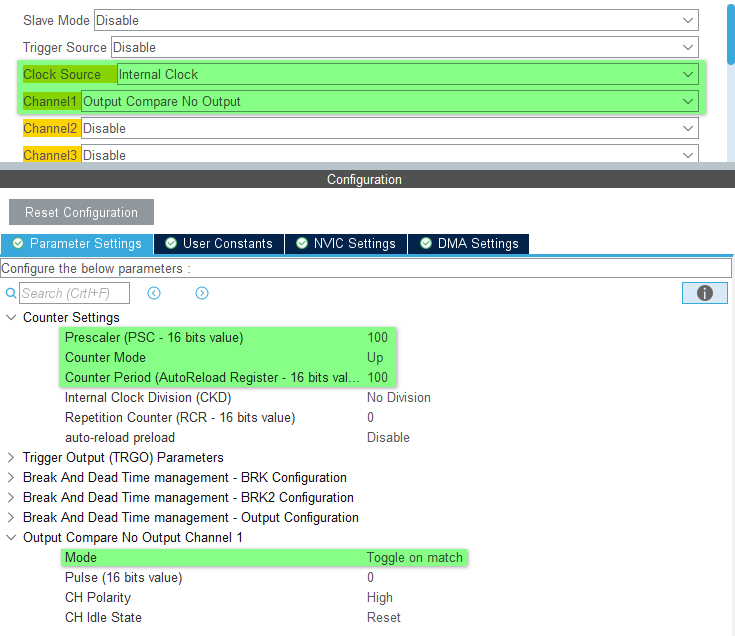
Sur la Figure 3 se trouve la configuration du Timer 1.

Figure - Configuration du Timer 1

# Configuration de l’ADC

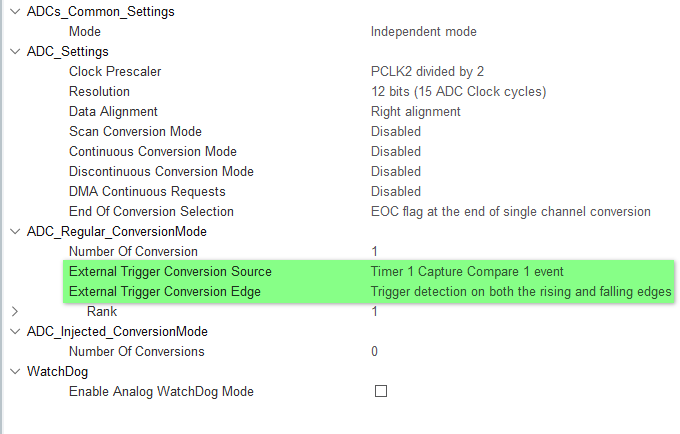
Nous avons configuré l’ADC de sorte qu’à chaque fois que le Timer 1 inverse sa sortie, une lecture ADC est lancée.

Figure - Configration de l'ADC

# Affichage

Pour l’affichage il a fallu coder une fonction doShowAnalogSignal() de la classe OscilloscopeController.

* Affichage du signal avec la fonction drawGraphPoints.
* Changement de l’axe du temps avec les boutons tactiles de l’écran
* Fonction de trigger

# Situation actuelle

Je n’ai pas eu le temps d’intégrer le RTOS à mon projet. J’ai préféré effectuer certaines des tâches supplémentaires.

|  |  |  |
| --- | --- | --- |
| Task | Résultat | Remarque |
| **Tâche 1 – Projet STM32CubeMX** | Fonctionnel |  |
| **Tâche 2 – Éteindre USB, FatFS et Ethernet** | Fonctionnel |  |
| **Tâche 3 – Configuration du RCC** | Fonctionnel |  |
| **Tâche 4 – Configuration de la génératrice de code**  **de CubeMX** | Fonctionnel |  |
| **Tâche 5 – Build et Debug** | Fonctionnel |  |
| **Tâche 6 – Configuration du Timing** | Fonctionnel |  |
| **Tâche 7 – Générateur de Fréquence Externe** | Fonctionnel |  |
| **Tâche 8 – Configuration de l'ADC (Software Triggered)** | Fonctionnel |  |
| **Tâche 9 – Configuration du timer Hardware (TIM)** | Fonctionnel |  |
| **Tâche 10 – Configuration du ADC (Timer Triggered)** | Fonctionnel |  |
| **Tâche 11 – XF Integration** | Fonctionnel |  |
| **Tâche 12 – Application** | Fonctionnel |  |
| **Tâche 13 – Oscilloscope GUI** | Fonctionnel |  |
| **Tâche 14 – Sample-Rate Tuning** | Fonctionnel |  |
| **Tâche 15 – RTOS Integration** | Non effectué | temps |
| **FA1 : Fonction Trigger** | Fonctionnel |  |
| **FA2: Display – Axe du temps** | Fonctionnel |  |
| **FA3 : Enregistrer les valeurs échantillonnées à l'aide du DMA** |  | Non obligatoire |

# Questions

## Configuration du timing

*Est-ce qu'il est possible d'exécuter le composant numéro 1 avec un XF ou bien faut-il un RTOS ? Justifiez votre réponse.*

* Non, il est impossible d’exécuter le composant numéro 1 avec un XF ou un RTOS.

*Est-ce qu'il est possible d'exécuter le composant numéro 2 avec un XF ou bien faut-il un RTOS ? Justifiez votre réponse.*

* Oui, car ce sont des fréquences plus basses.

*Si l'on combine un timer hardware avec un XF, lequel des deux doit être priorisé ? Justifiez votre réponse.*

* Le timer hardware, il doit arrêter le programme un instant pour effectuer sa ISR.

## Configuration de l’ADC

*Combien de mesures [Samples/s] le convertisseur A/D doit-il effectuer par seconde pour pouvoir échantillonner des signaux avec des fréquences jusqu'à 1 kHz ?*

* Théorème de nyquist : fs = finmin \* 2

*Faut-il un filtre ? Si oui, quelle sera la fréquence de coupure de ce filtre ?*

* Il faut un filtre anti-aliasing de 1kHz.

*Est-ce que la fréquence donnée par le théorème d’échantillonnage ou devrait-elle être plus élevée ?*

* Plus elle est élevée plus le signal reconstruit sera proche du signal échantillonné

*Lequel des canaux du ADC3 doit être utilisé pour pouvoir mesurer / échantillonner le signal l'aide de la broche PA0 ?*

* Le canal IN0

Est-ce que le ADC pourrait-il éventuellement effectuer des mesures à des intervalles réguliers à l'aide de ses propres moyens ?

* Non, uniquement en continu. Il faut un timer pour que ce soit précis.

## Sample-Rate Tuning

*Quelle fréquence d’échantillonnage peut être atteinte ?*

* *20kHz*

*Quel(s) composant(s) limite(nt) le système?*

Au bout d’un moment, il n’y a que des interruptions et le XF n’arrive plus a être exécuté correctement. Il faut alors un DMA qui s’occupe de ca en parallèle.

# Annexes

Figure - Périphériques cadencés par APB2